动手一天学深度学习

1 基础 · 2 卷积网络 · 3 计算 · **4 计算机视觉**

深度学习实训营 2019

何通，李沐

http://1day-zh.d2l.ai

aws

大纲

- 图像增广
- 微调
- 目标检测
- 边界框和锚框
- 单发多框检验（SSD）模型
- 只看一次（YOLO）
- 区域卷积神经网络（R-CNN）
- 计算机视觉训练技巧

图像增广

p=1.0

size_percent=0.30

p=0.50

cutoff=0.00

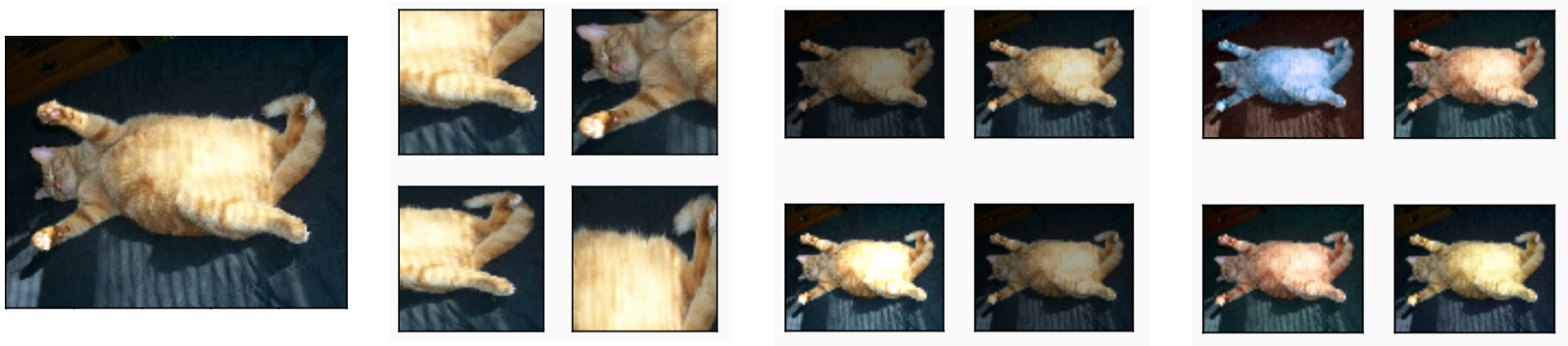
http://1day-zh.d2l.ai

CES'19 上的真实故事

- 一家初创公司要演示一款智能自动售货机，通过相机镜头识别客户挑选的物品。
- 但演示失败了，因为展厅有不同的：
  - 光照色温
  - 桌子上有光反射
- 他们通宵加班：
  - 重新收集数据并训练新模型
  - 订购了桌布

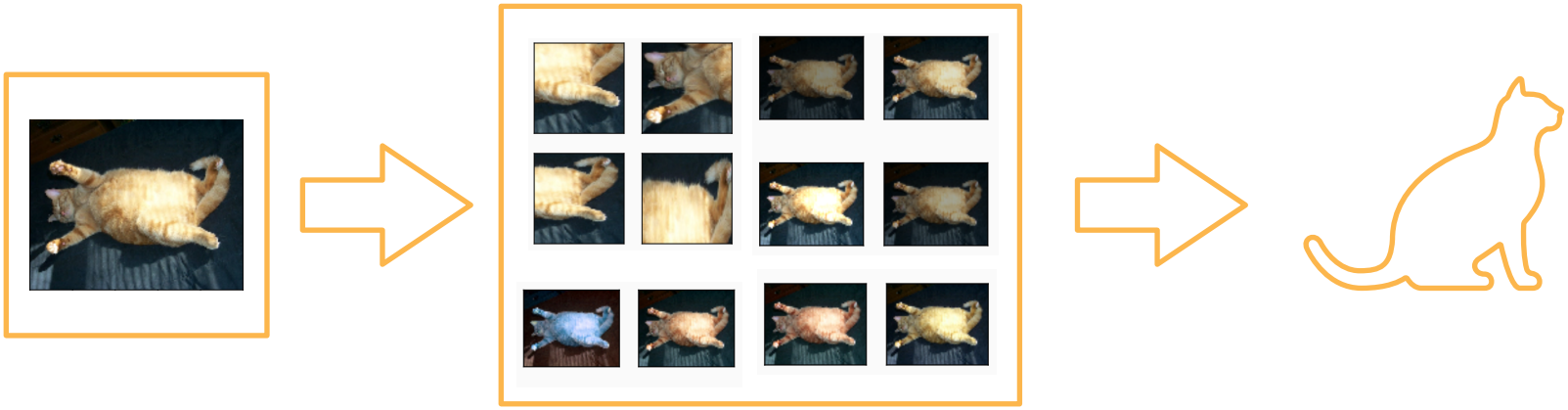
数据增广

- 现有数据集基础上，增广为具有更多多样性的数据集：
- 在训练数据集中添加各种背景噪音
- 转换为其他图像：改变颜色，改变形状



http://1day-zh.d2l.ai

用增广数据训练



原始数据集

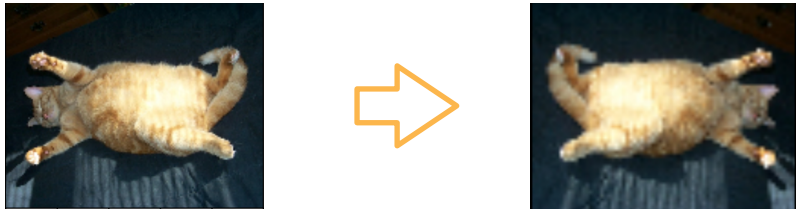
增广后数据集

模型

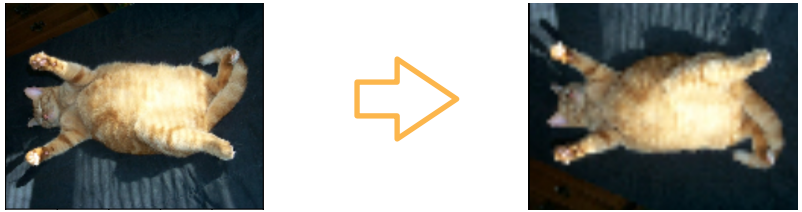
http://1day-zh.d2l.ai

翻转

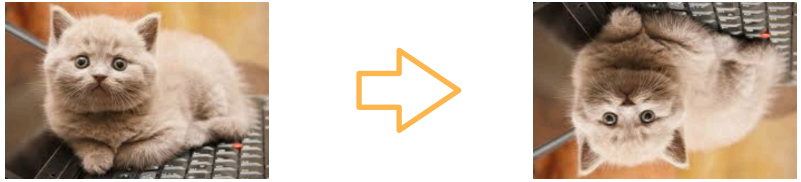
- 左右翻转



- 上下翻转



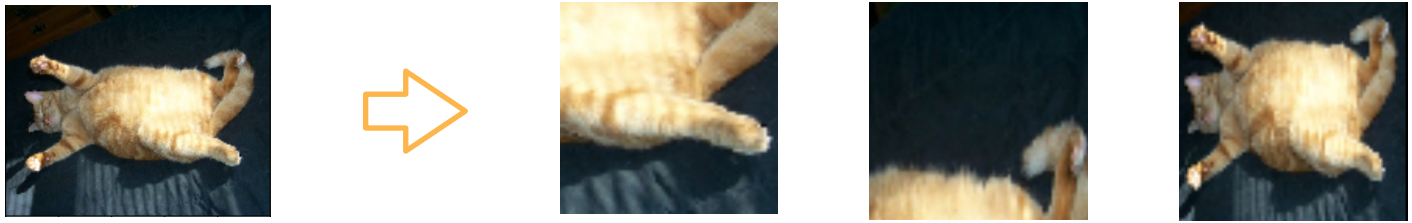
- 不一定都有效



http://1day-zh.d2l.ai

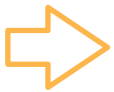
裁剪

- 从图像中随机裁剪一块区域，然后调整其大小
  - 随机宽高比（例如 [3/4, 4/3]）
  - 随机的区域
  - 随机区域面积大小（例如 [8%, 100%]）



变色

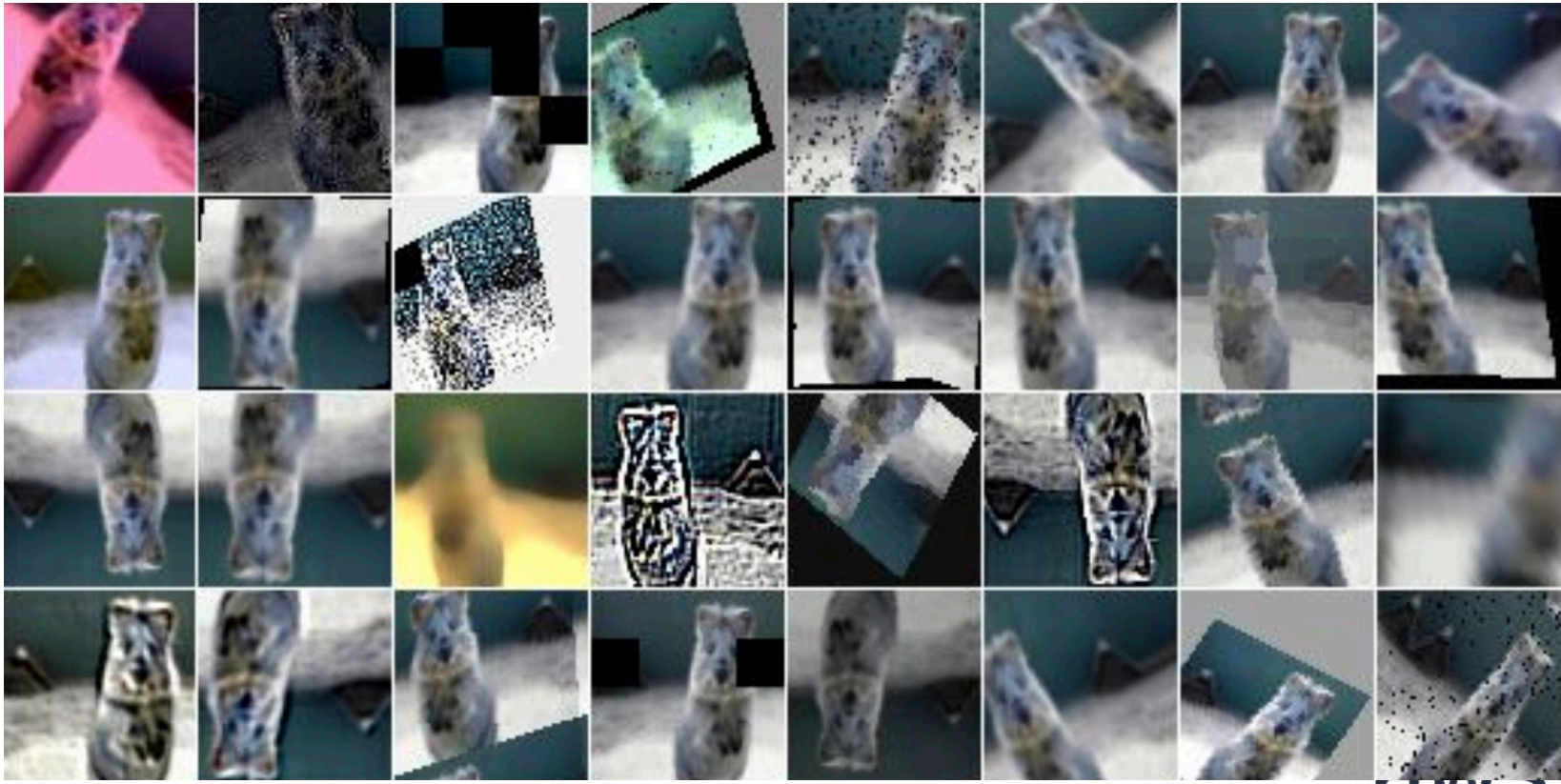
- 调整色调，饱和度和亮度 （例如 [0.5, 1.5]）



亮度

色调

更多的增广方法



https://github.com/aleju/imgaug

http://1day-zh.d2l.ai

微调

http://1day-zh.d2l.ai

标记数据集非常昂贵

流行数据集

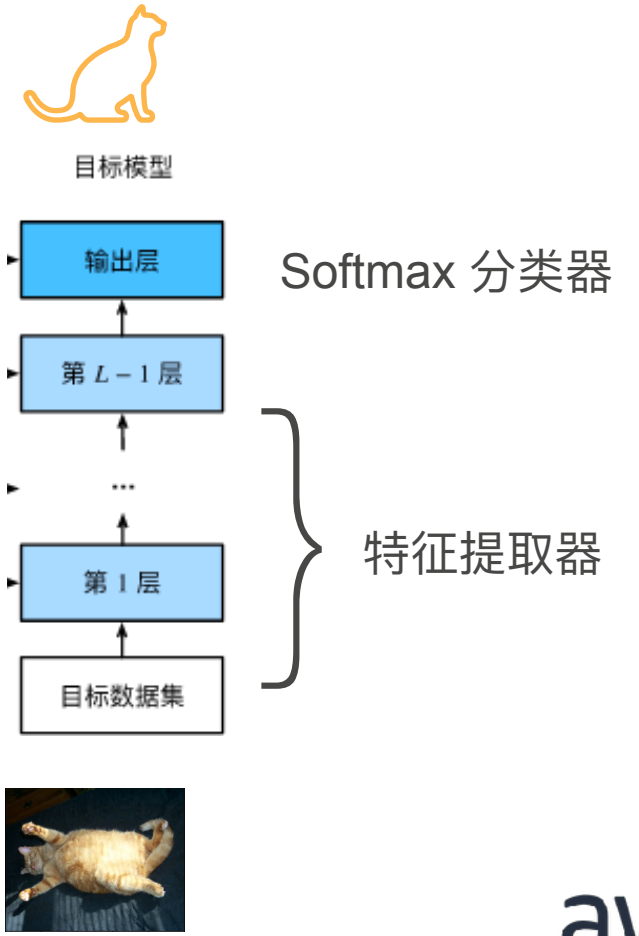
# 数据点	1.2 M	50K	60 K
# 类别	1,000	100	10

http://1day-zh.d2l.ai

aws

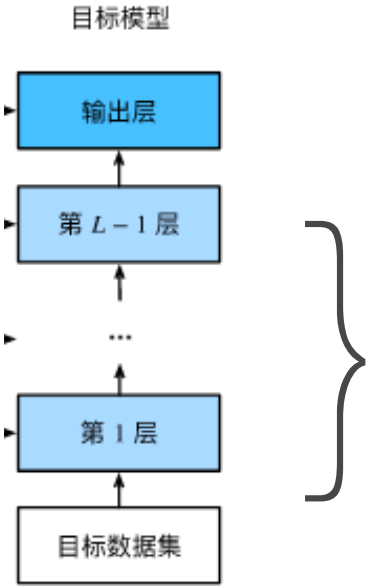
网络结构

- 一个神经网络模型可以大致分为两部分
  - 特征提取器将原始像素映射为线性可分离的特征
  - 用线性分类做决定



http://1day-zh.d2l.ai

微调



不直接使用原分类器的最后一层参数，因为标签可能不一致

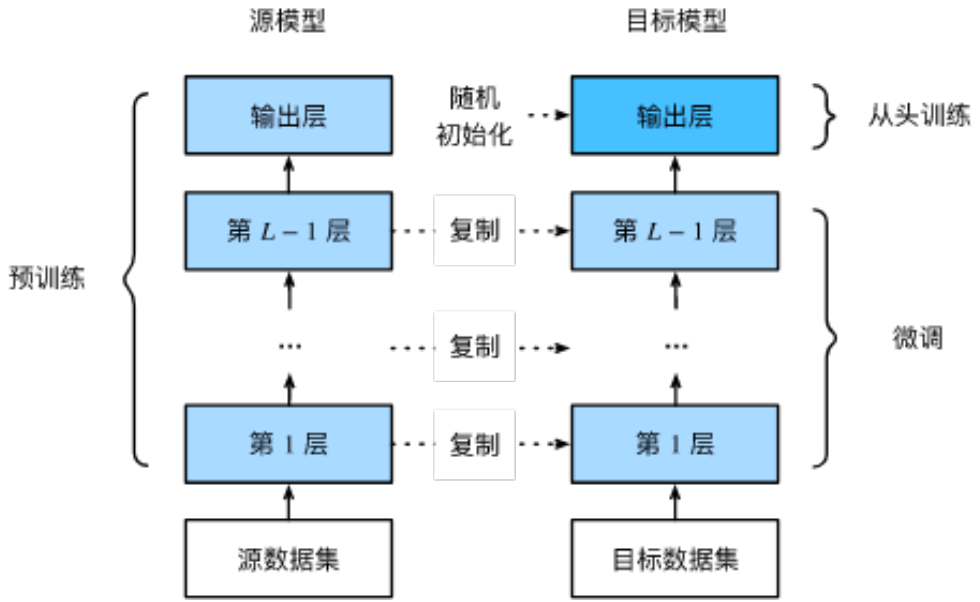
可能仍然是很好的图像特征提取器

源数据集

目标数据集

http://1day-zh.d2l.ai

微调中的权重初始化



源数据集

目标数据集

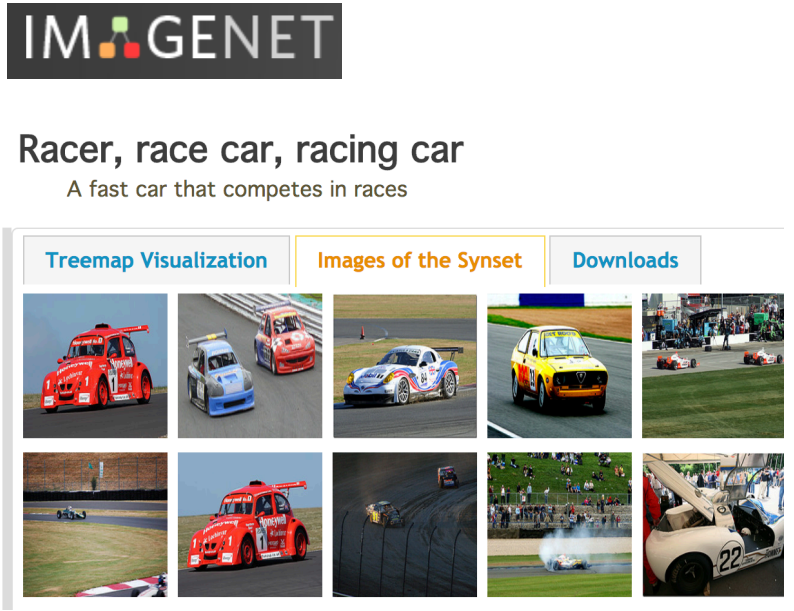
http://1day-zh.d2l.ai

微调训练

- 在已有模型的基础上微调到目标数据集
  - 使用较小的学习率
  - 使用较少的迭代周期
- 如果源数据集比目标数据集更复杂，则微调通常会得到更高质量的模型

重用已有分类器的参数

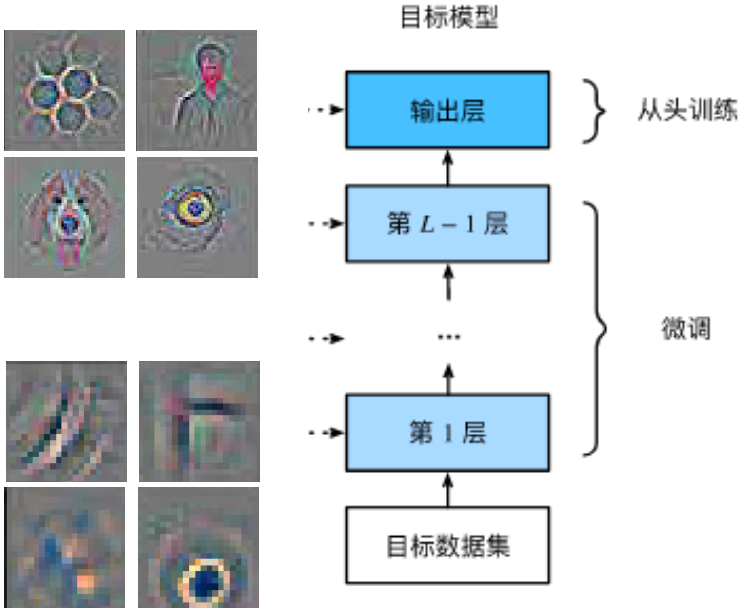
- 源数据集可能包含目标数据集中的某些类别
- 在初始化期间使用来自预训练模型的相应权重向量



http://1day-zh.d2l.ai

固定一些层的参数

- 神经网络学习分层特征表示
  - 低层特征是通用的
  - 高层特征与数据集中的对象更相关
- 在微调期间固定低层的参数
  - 一个有用的正则项



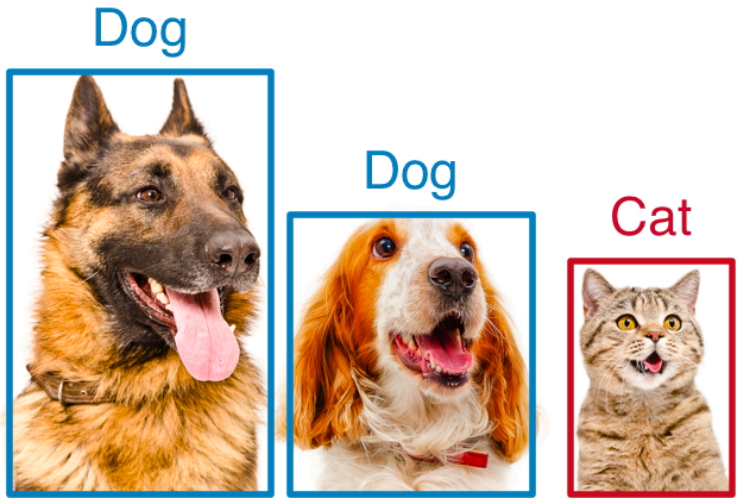
http://1day-zh.d2l.ai

目标检测

http://1day-zh.d2l.ai

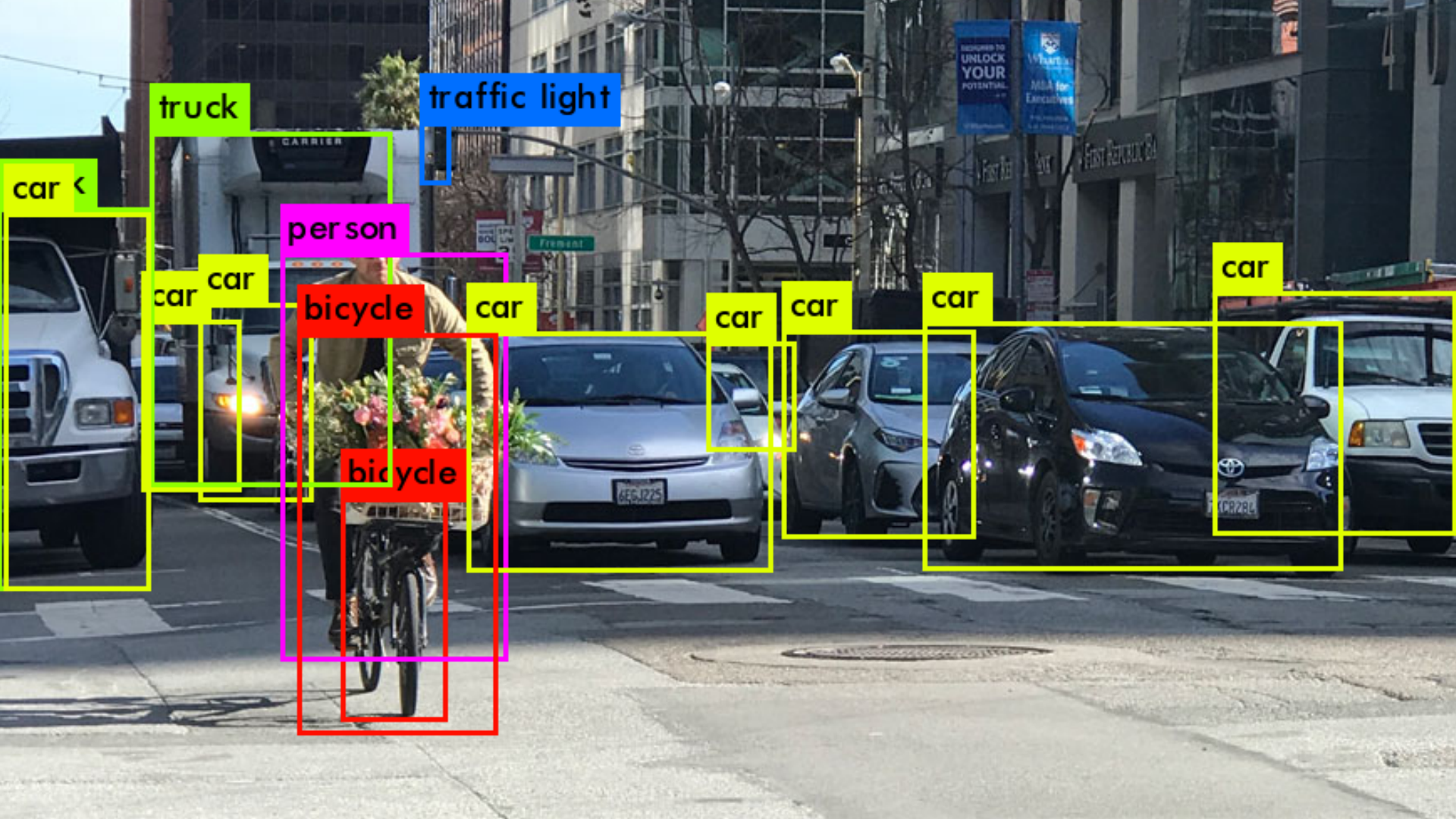
图像分类

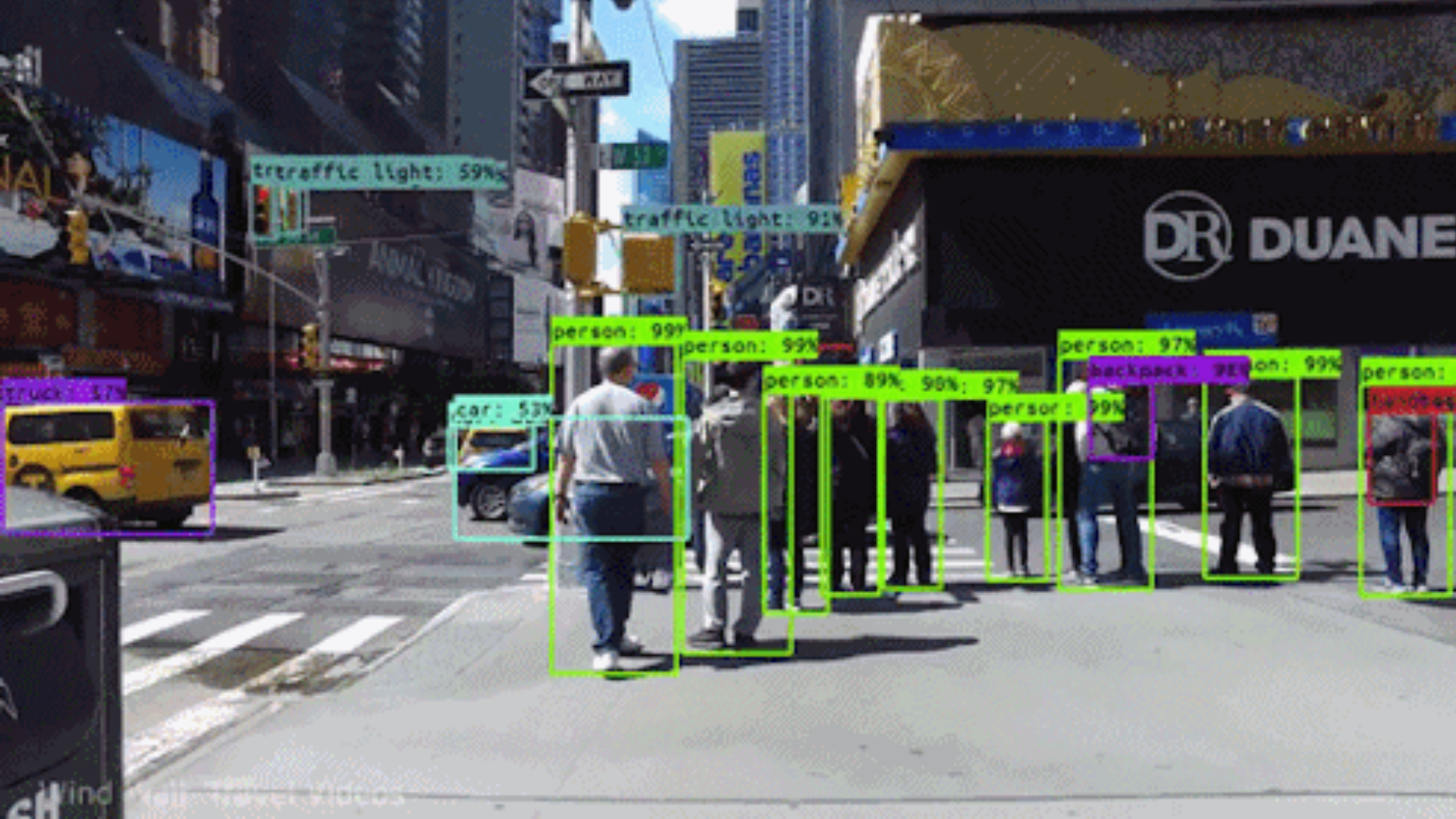
目标检测



http://1day-zh.d2l.ai

aws


truck
traffic light
car
person
car
car
bicycle
car
car
car
car
car
bicycle

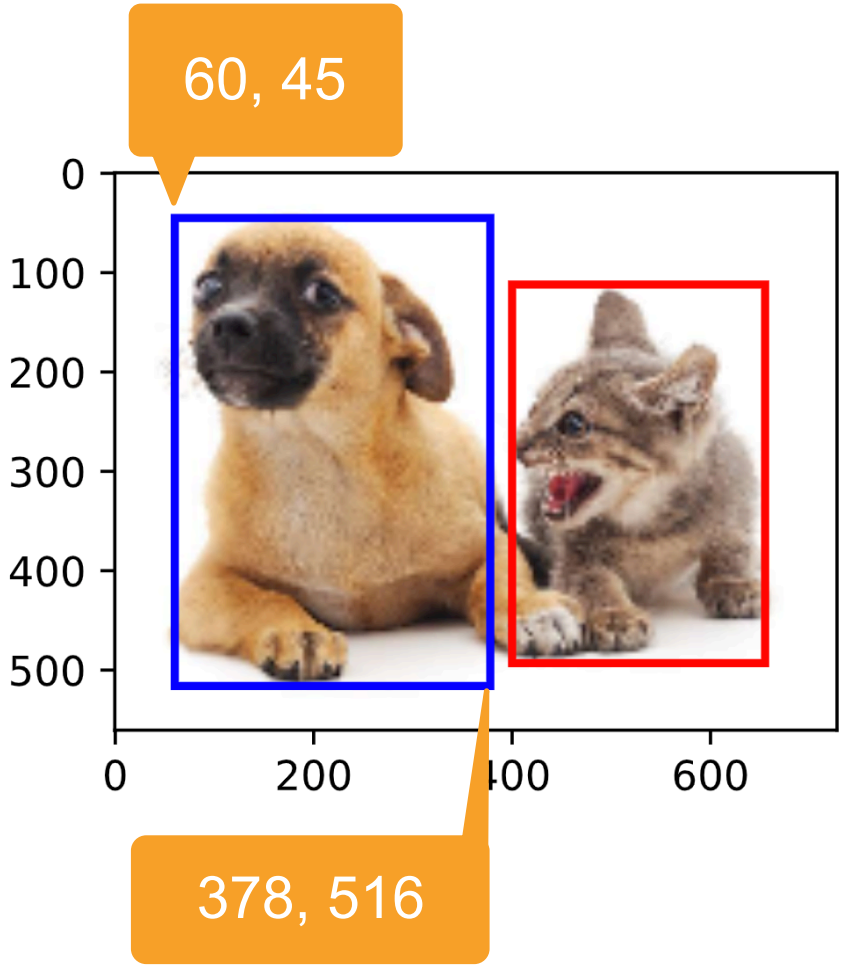

DUANE
traffic light: 59%
traffic light: 91%
person: 99%
person: 99%
person: 89%
person: 98%
person: 97%
person: 97%
backpack:
person: 99%
person:
car: 53%

边界框和锚框


prime
entrega gratis y rápida
streaming de video
prime
entrega gratis y rápida
streaming de video
prime
entrega gratis y rápida
prime

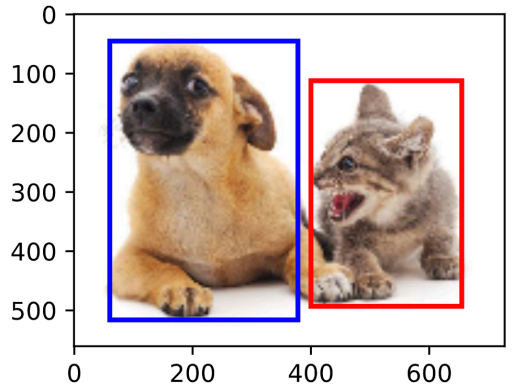
边界框

- 用4个数定义一个边界框
  - (左上 x, 左上 y, 右下 x, 右下 y)
  - (左上 x, 左上 y, 宽, 高)

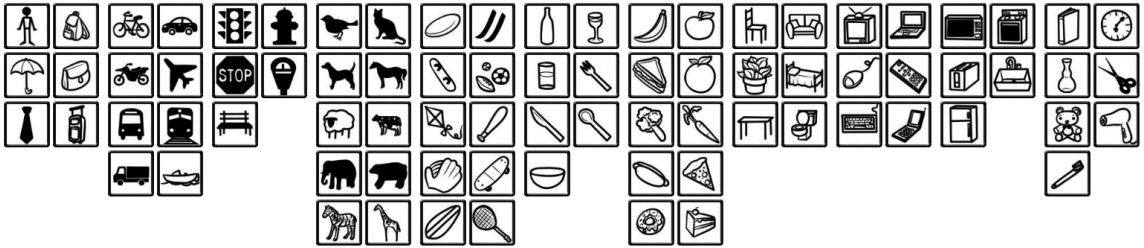


http://1day-zh.d2l.ai

目标检测数据集



- 每行都有一个目标物体
  - 图像名称，物体类别，边界框
- COCO (cocodataset.org)
  - 80 类物体
  - 330K 张图
  - 1.5M 个目标物体

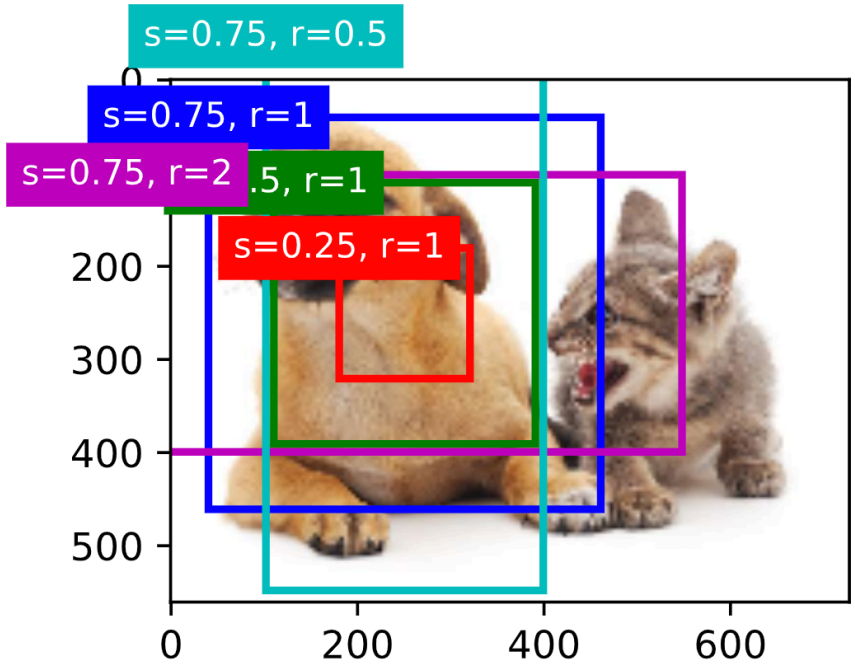


http://1day-zh.d2l.ai

aws

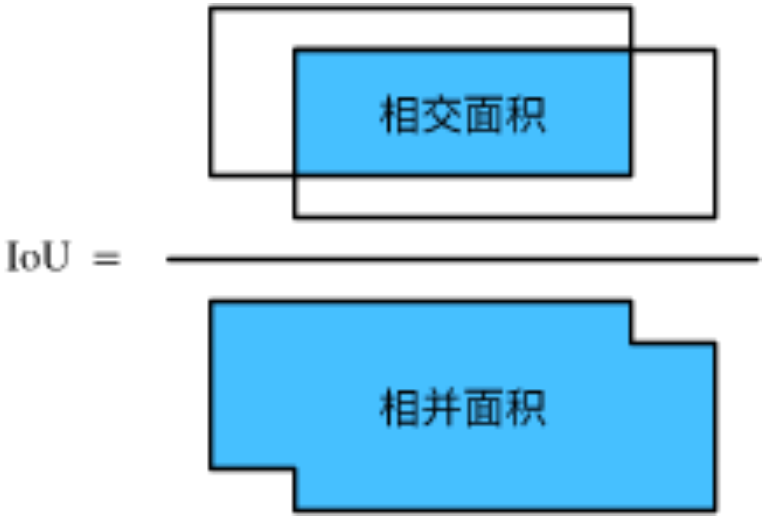
锚框

- 目标物体检测算法
  - 选择多个区域，称为锚框
  - 预测每个锚框是否包含目标物体
  - 如果是，则再预测从锚框到实际边界框的偏移量



交并比

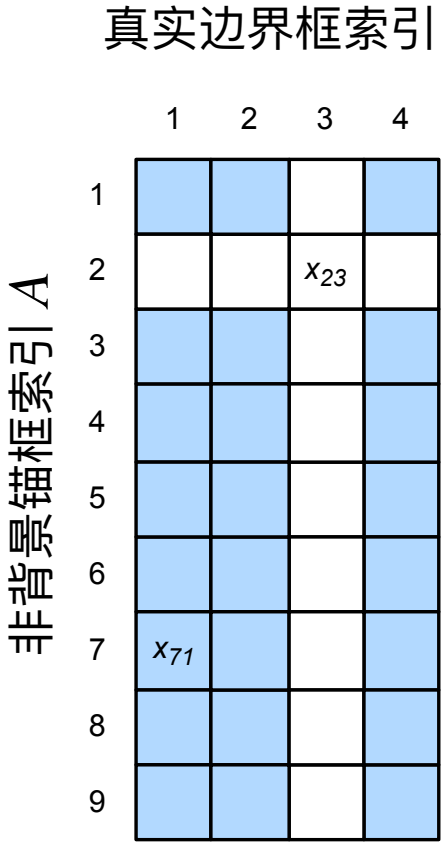
- 交并比（IoU）测量两个框之间的相似性：
  - 0 表示不重叠
  - 1 表示完全相同



将标签分配给锚框

- 每个锚框都是一个训练样例
- 标记每个锚框的内容为
  - 背景，或
  - 与某个边界框有关
- 可能会生成大量的锚框
  - 大部分锚框都是背景框

将标签分配给锚框

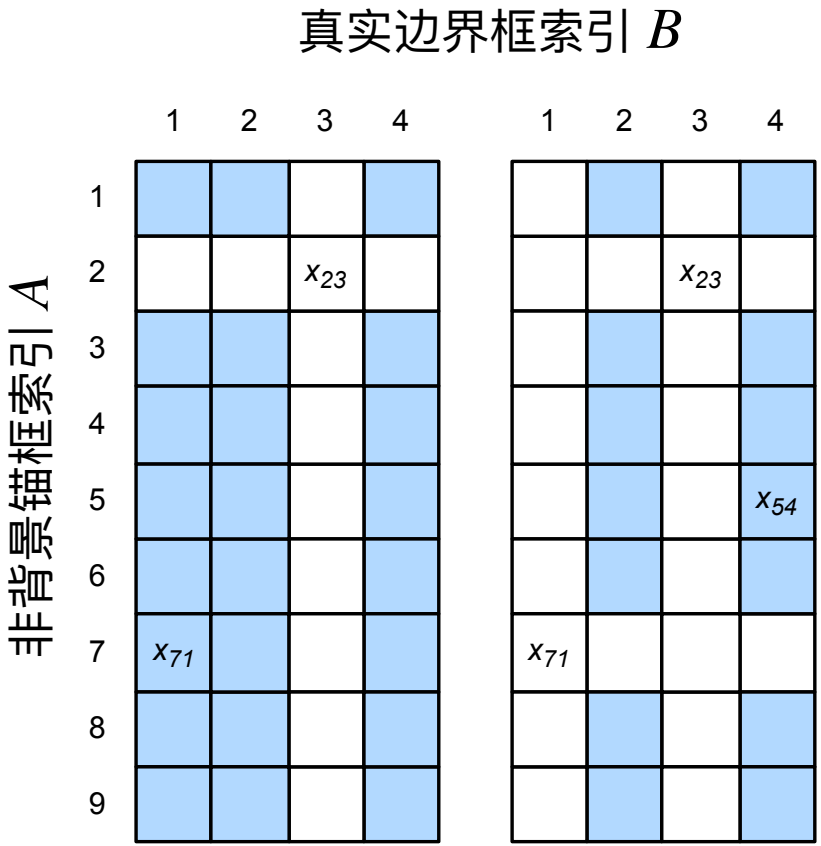


假设IoU矩阵 X 中最大值为 x_{23}， 我们将分配真实边界框 B_3 给锚框 A_2。

然后，丢弃矩阵中第2行和第3列的所有元素， 找出剩余阴影部分的最大元素 x_{71}，为锚框 A_7 分配真实边界框 B_1。

http://1day-zh.d2l.ai

将标签分配给锚框

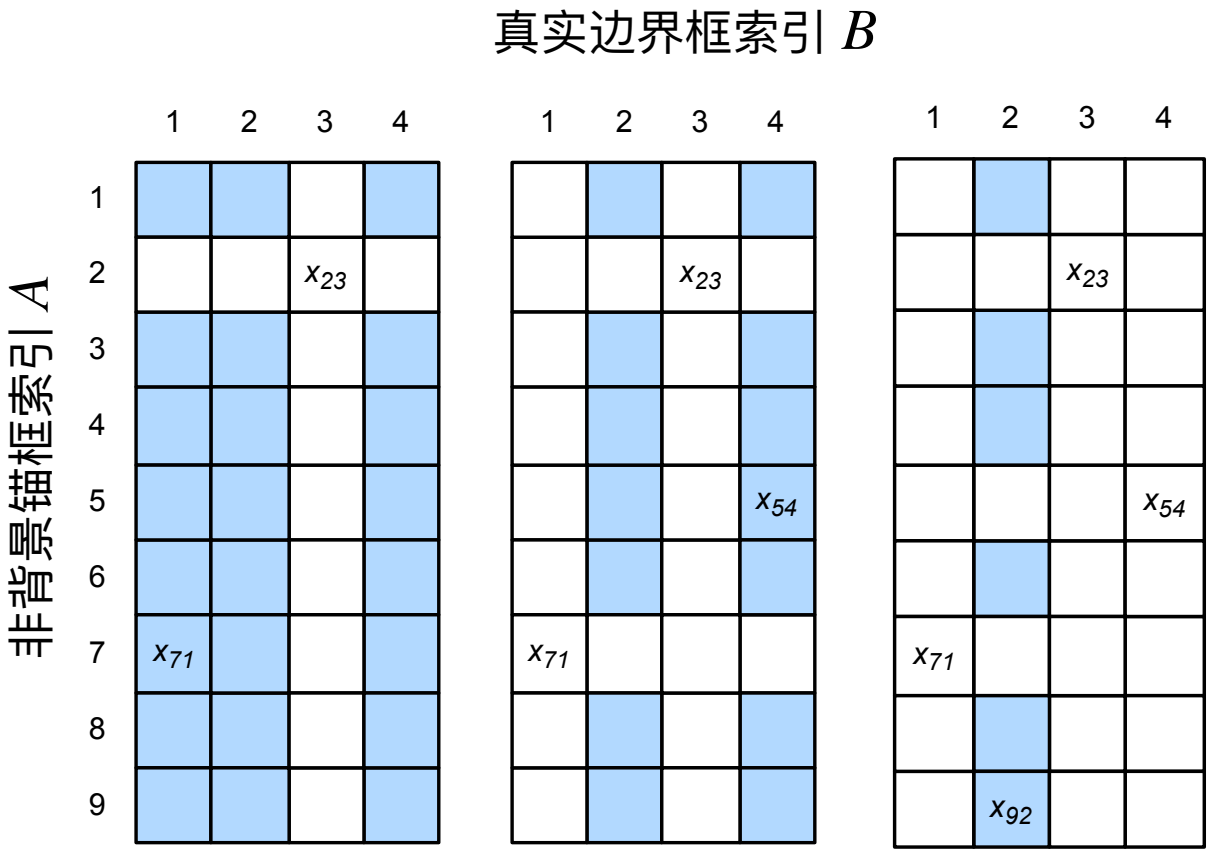


接着，丢弃矩阵中第7行和第1列的所有元素，

找出剩余阴影部分的最大元素 x_{54}，

为锚框 A_5 分配真实边界框 B_4 。

将标签分配给锚框

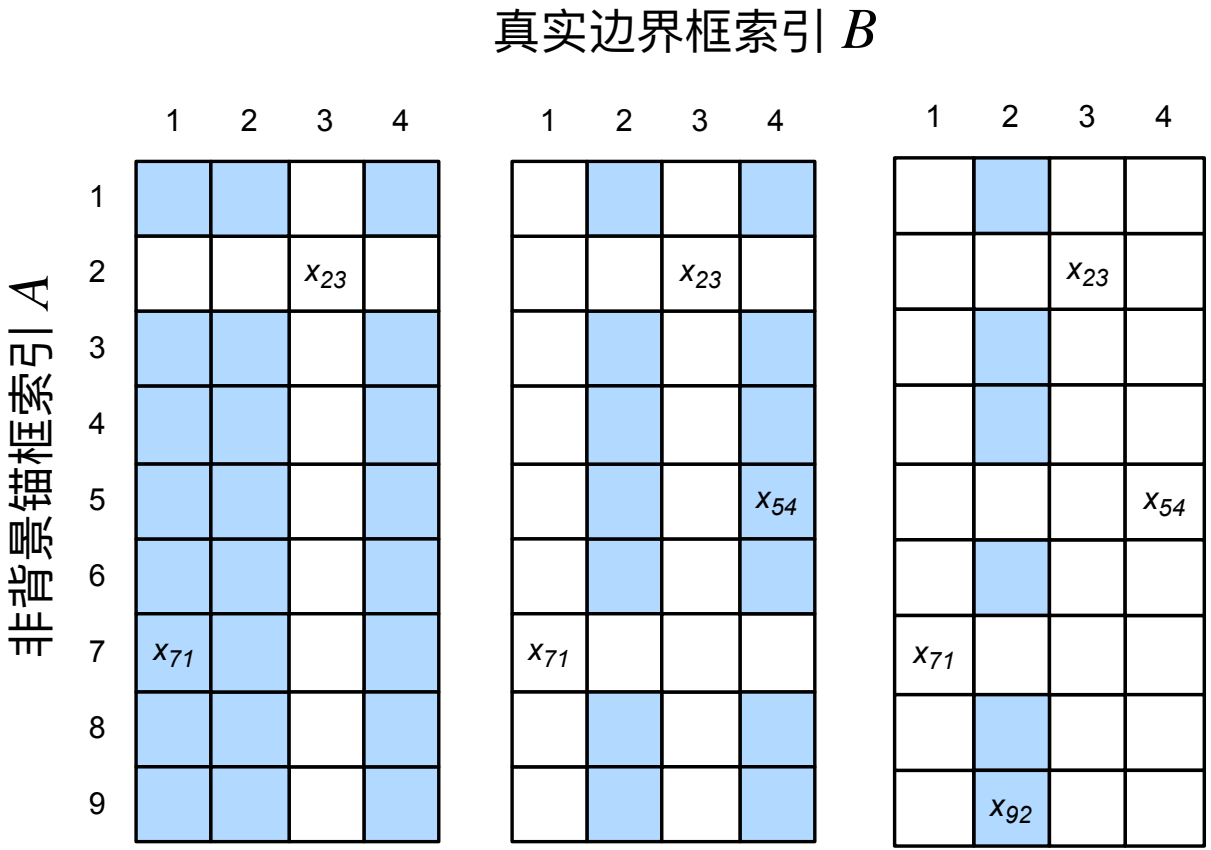


最后，丢弃矩阵中第5行和第4列的所有元素，

找出剩余阴影部分的最大元素 x_{92}，

为锚框 A_9 分配真实边界框 B_2 。

将标签分配给锚框



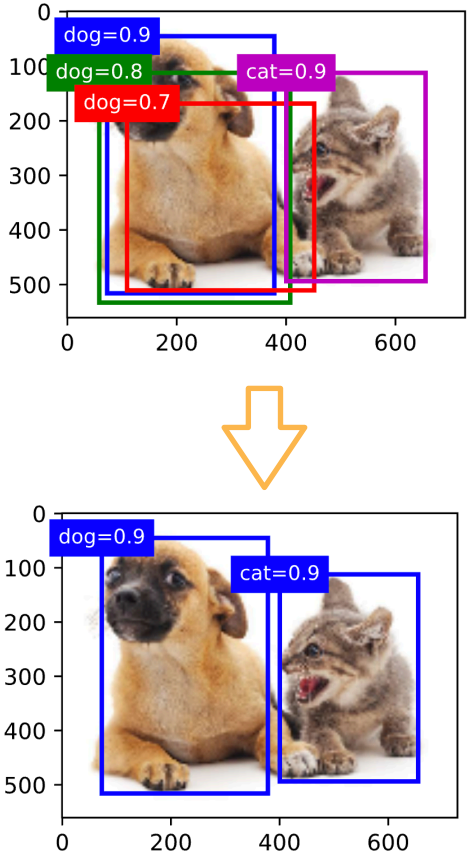
我们遍历除去 A_2 A_5 A_7 A_9 的每个剩余锚框

- 判断与真实边界框最大的IoU是否超过阈值
- 如果是，为其分配对应的真实边界框

http://1day-zh.d2l.ai

输出预测边界框（NMS）

- 非极大值抑制（non-maximum suppression, NMS）
  - 每个有配对锚框生成一个概率预测
  - 选中得分最高的那个预测 P
  - 计算所有其他预测与P的IoU，将值大于 θ 的预测删除
  - 重复上述过程，直到所有锚框都被选中或者删除

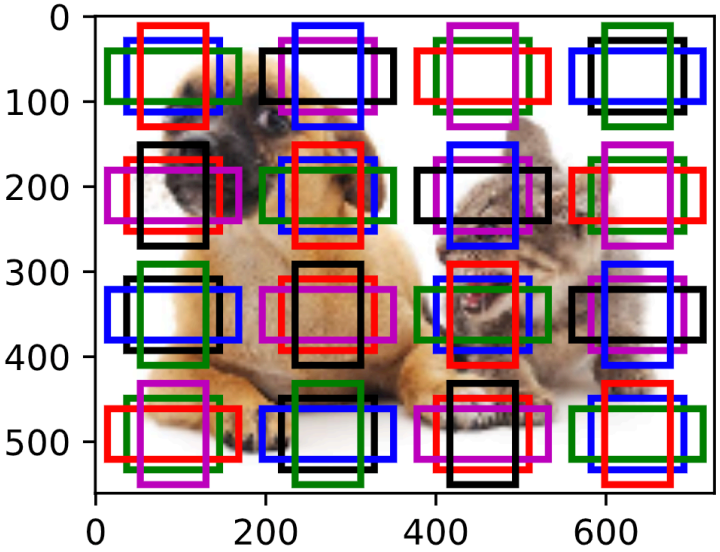


http://1day-zh.d2l.ai

单发多框检测（SSD）

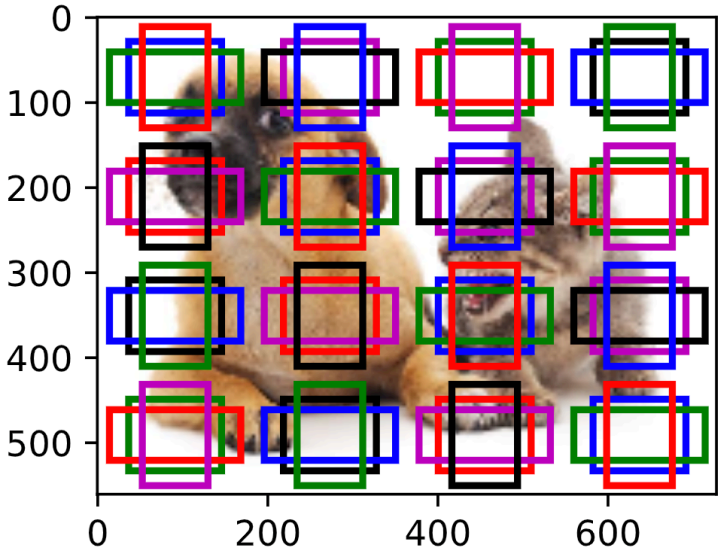
生成锚框

- 中心与比例确定一个锚框
  - 每个像素都可以成为中心
  - 多个可能的长宽比



多尺度目标检测

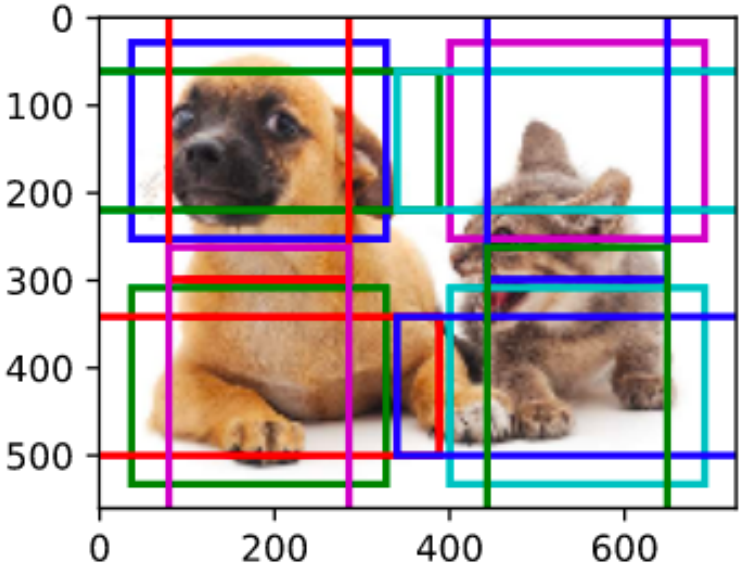
- 每个像素都生成锚框
  - 锚框数量太多
  - 区域重复
- 多尺度生成锚框
  - 小尺度检测小目标
  - 大尺度检测大目标



http://1day-zh.d2l.ai

多尺度目标检测

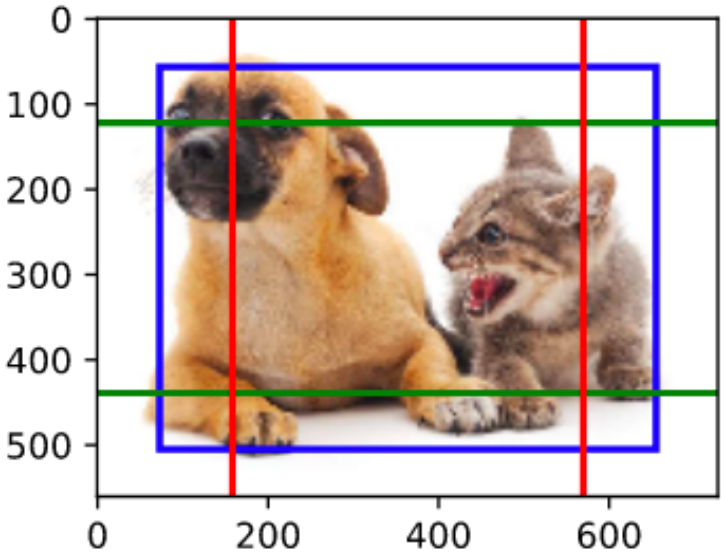
- 每个像素都生成锚框
  - 锚框数量太多
  - 区域重复
- 多尺度生成锚框
  - 小尺度检测小目标
  - 大尺度检测大目标



http://1day-zh.d2l.ai

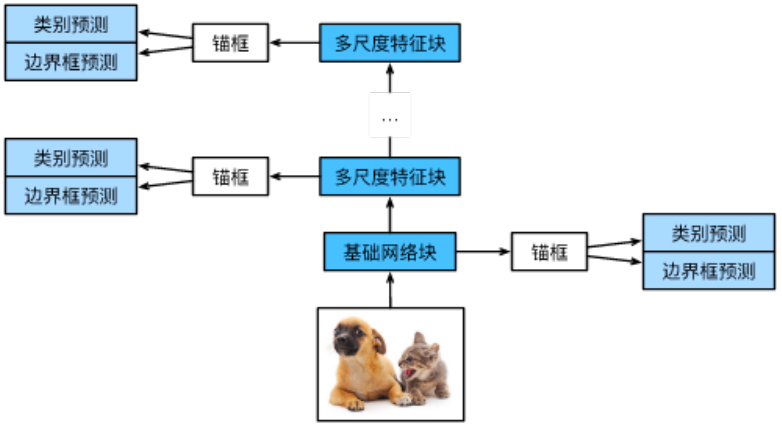
多尺度目标检测

- 每个像素都生成锚框
  - 锚框数量太多
  - 区域重复
- 多尺度生成锚框
  - 小尺度检测小目标
  - 大尺度检测大目标

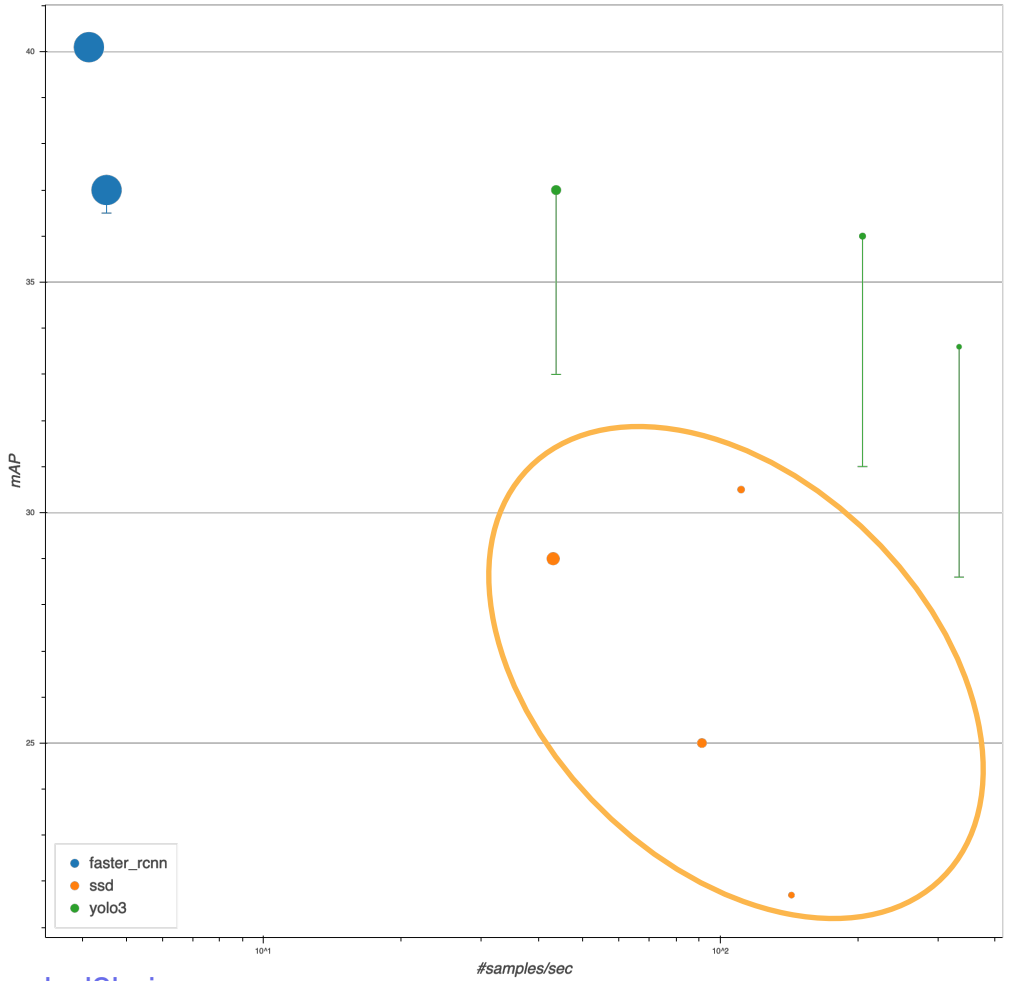


单发多框检测（SSD）模型

- 单发多框检测是一个多尺度的目标检测模型
  - 先是基础网络从原始图像中抽取特征，使它输出的高和宽较大，可以用来检测尺寸**较小**的目标
  - 接下来的每个多尺度特征块将上一层提供的特征图的高和宽缩小（如减半），并使特征图中每个单元在输入图像上的感受野变得更广阔
  - 越靠近顶部的多尺度特征块输出的特征图越小，故而基于特征图生成的锚框也越少，加之特征图中每个单元感受野越大，因此更适合检测尺寸**较大**的目标预测每个锚框的类和边界框



http://1day-zh.d2l.ai



https://gluon-cv.mxnet.io/model_zoo/detection.html

SSD

http://1day-zh.d2l.ai

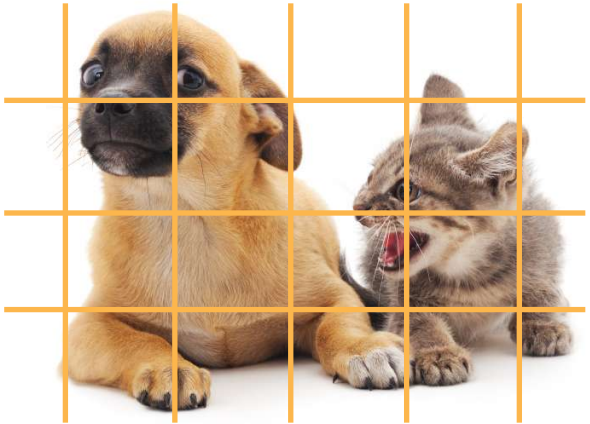
aws

只看一次 (YOLO)

http://1day-zh.d2l.ai

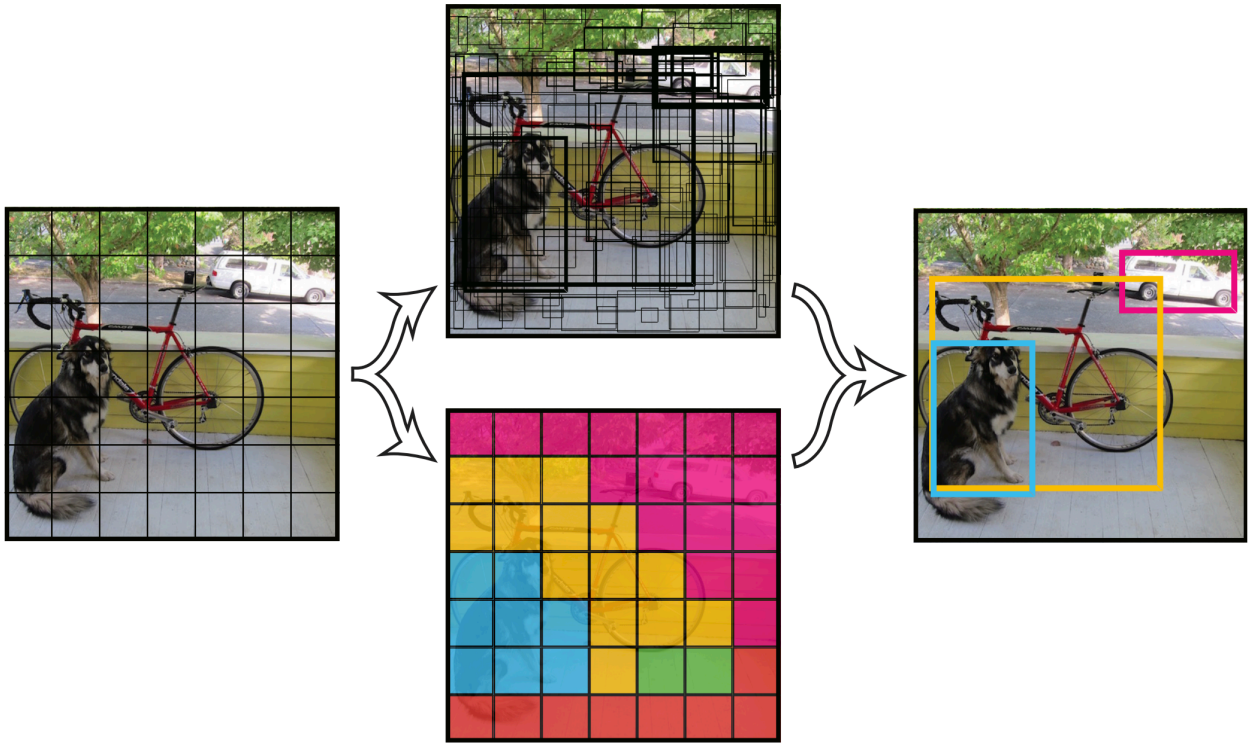
YOLO

- YOLO 将输入图像均匀地切割成 S x S 个小区域
- 每个小区域都作为中心生成一组锚框
- 真实边界框**只被分配到其中心点所在小区域生成的锚框**
- V2 和 V3 增加了更多改进

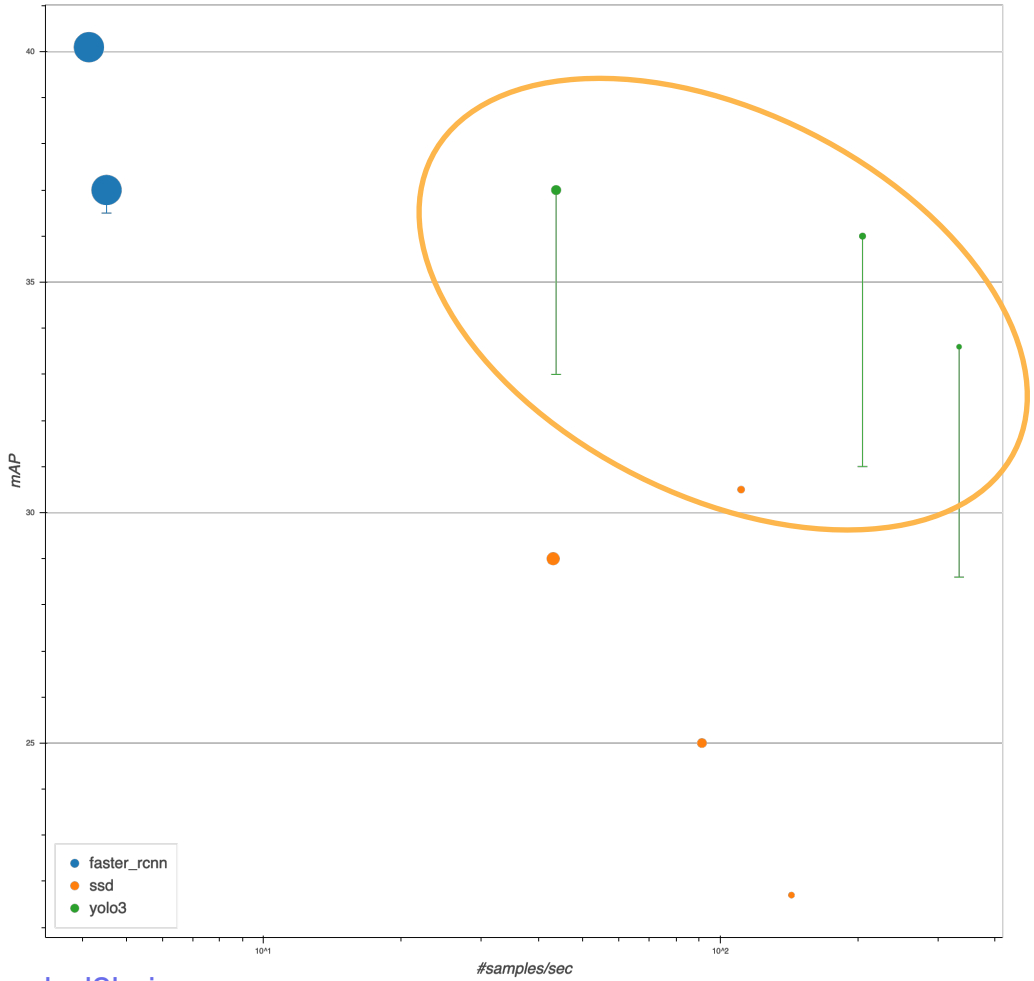


http://1day-zh.d2l.ai

YOLO



http://1day-zh.d2l.ai

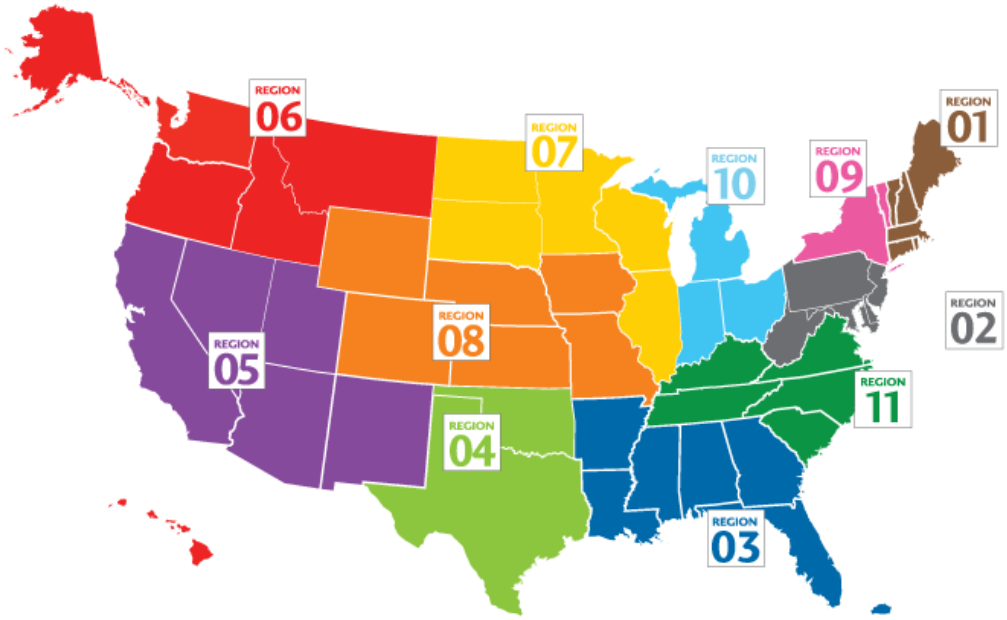


https://gluon-cv.mxnet.io/model_zoo/detection.html

Yolo V3

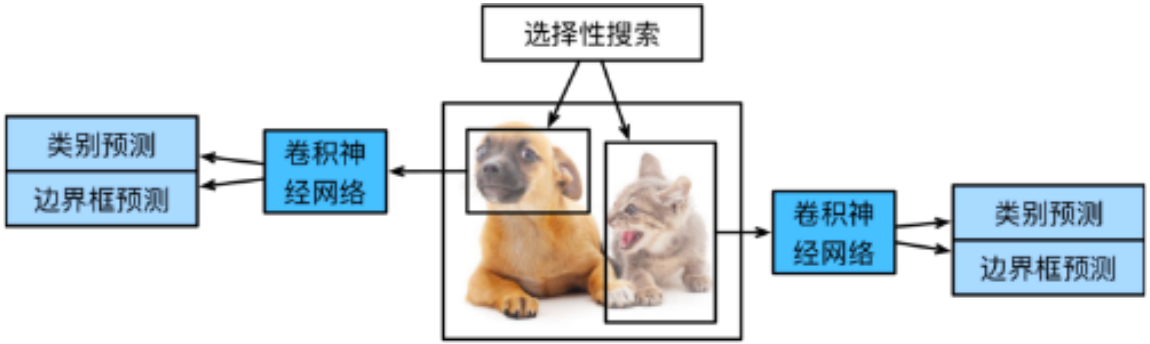
http://1day-zh.d2l.ai

区域卷积神经网络



http://1day-zh.d2l.ai

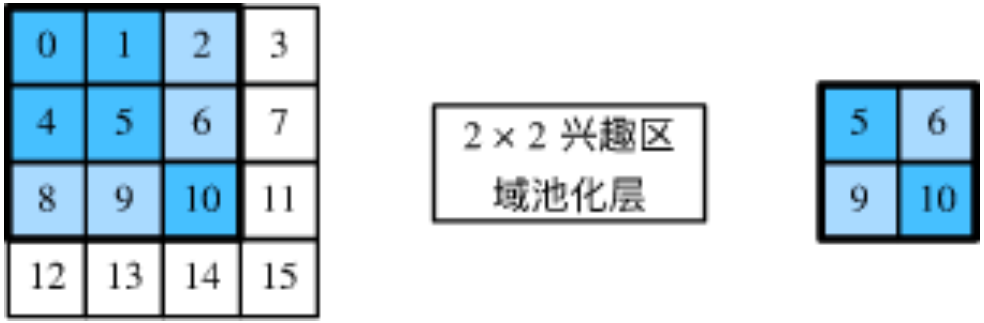
R-CNN



- 区域卷积神经网络（region-based CNN，R-CNN）
  - 对输入图像使用选择性搜索，来选取多个高质量的区域提议
  - 选取一个预训练的卷积神经网络，并通过前向计算输出抽取的提议区域特征
  - 将每个提议区域的特征连同其标注的类别作为一个样本，训练多个支持向量机对目标分类
  - 训练线性回归模型来预测真实边界框

http://1day-zh.d2l.ai

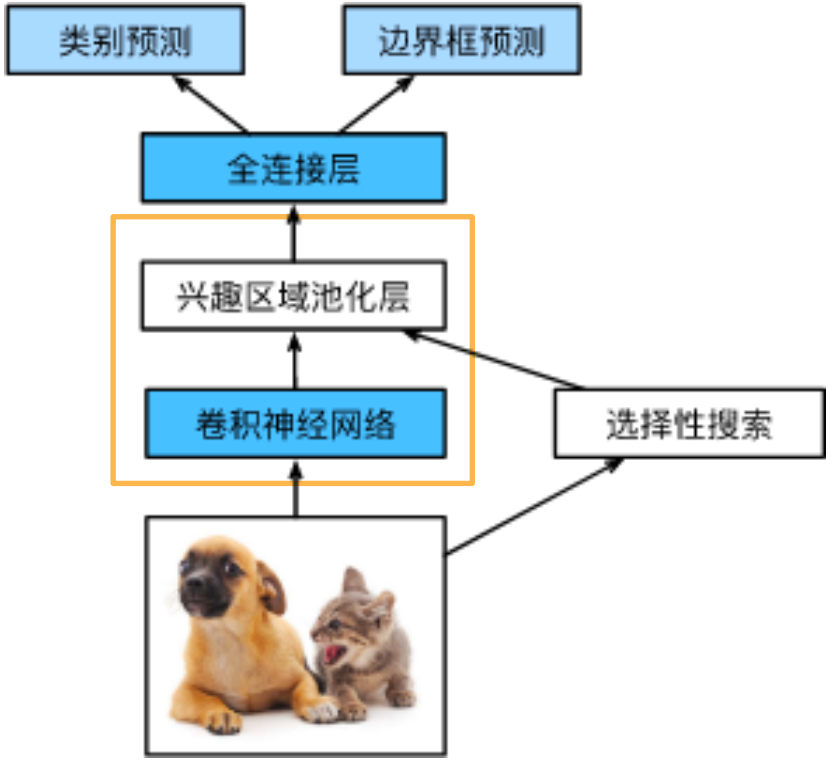
兴趣区域 (RoI) 池化层



- 兴趣区域池化层（region of interest pooling, RoI）池化层
  - 将一个区域均匀地切割成 n x m 个块，输出每个块中的最大值
  - 不同大小的区域都固定返回 n x m 个值

Fast RCNN

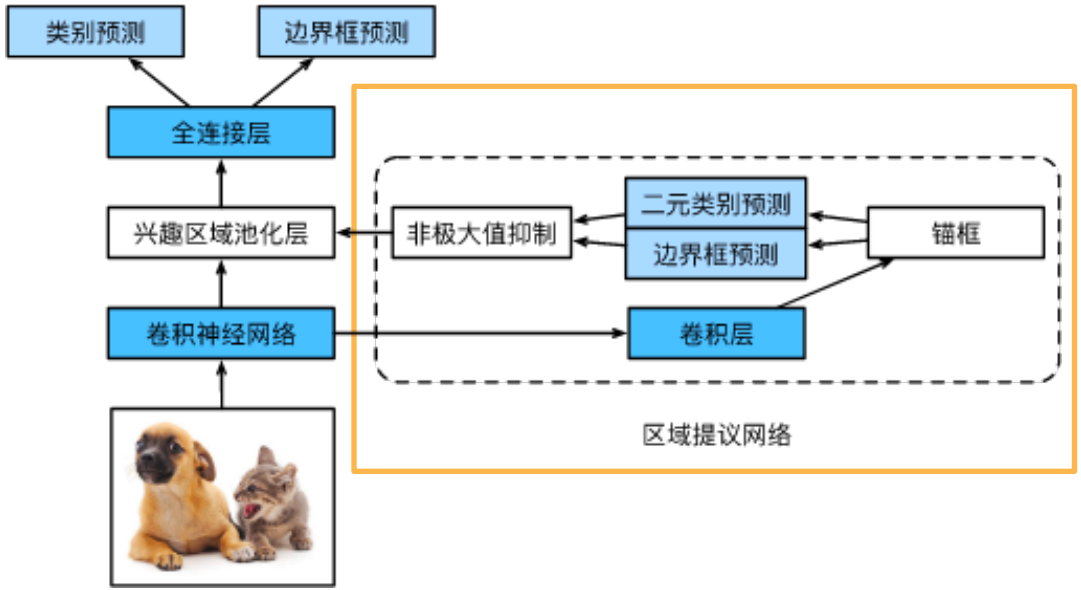
- 用CNN快速提取特征：
  - 输入是整个图像，而不是各个提议区域
  - RoI池化为每个区域返回固定大小的特征



http://1day-zh.d2l.ai

Faster R-CNN

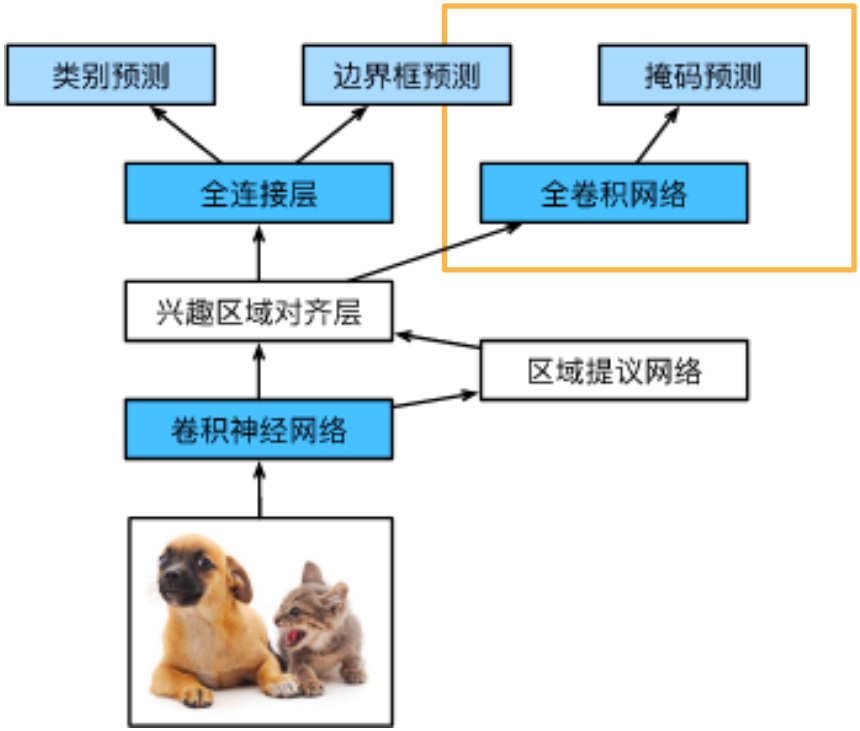
- 与 Fast R-CNN 相比，只有生成提议区域的方法从选择性搜索变成了区域提议网络，而其他部分均保持不变。



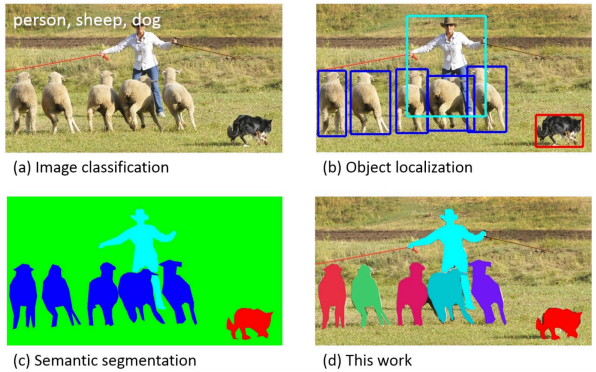
http://1day-zh.d2l.ai

Mask R-CNN

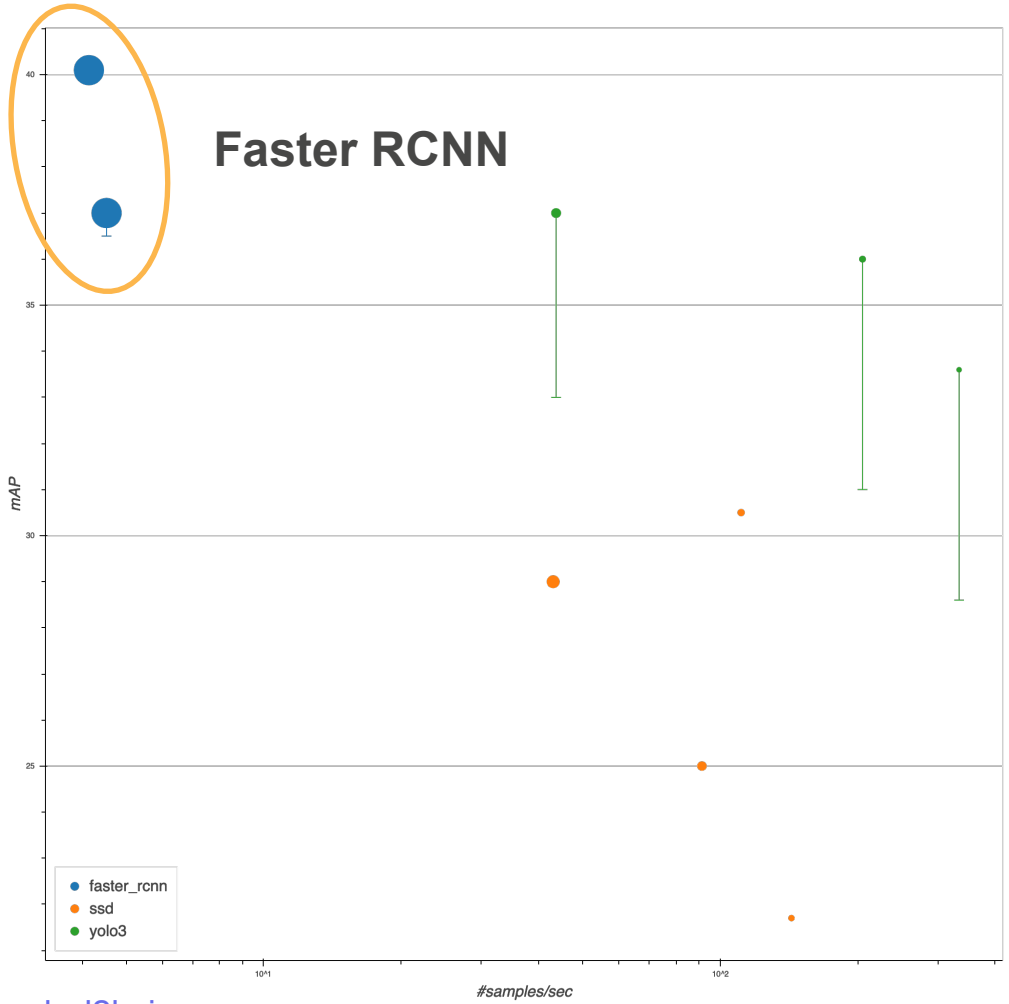
- 如果训练数据还标注了每个目标在图像上的像素级位置，那么Mask R-CNN 能有效利用这些详尽的标注信息进一步提升目标检测的精度



COCO



http://1day-zh.d2l.ai



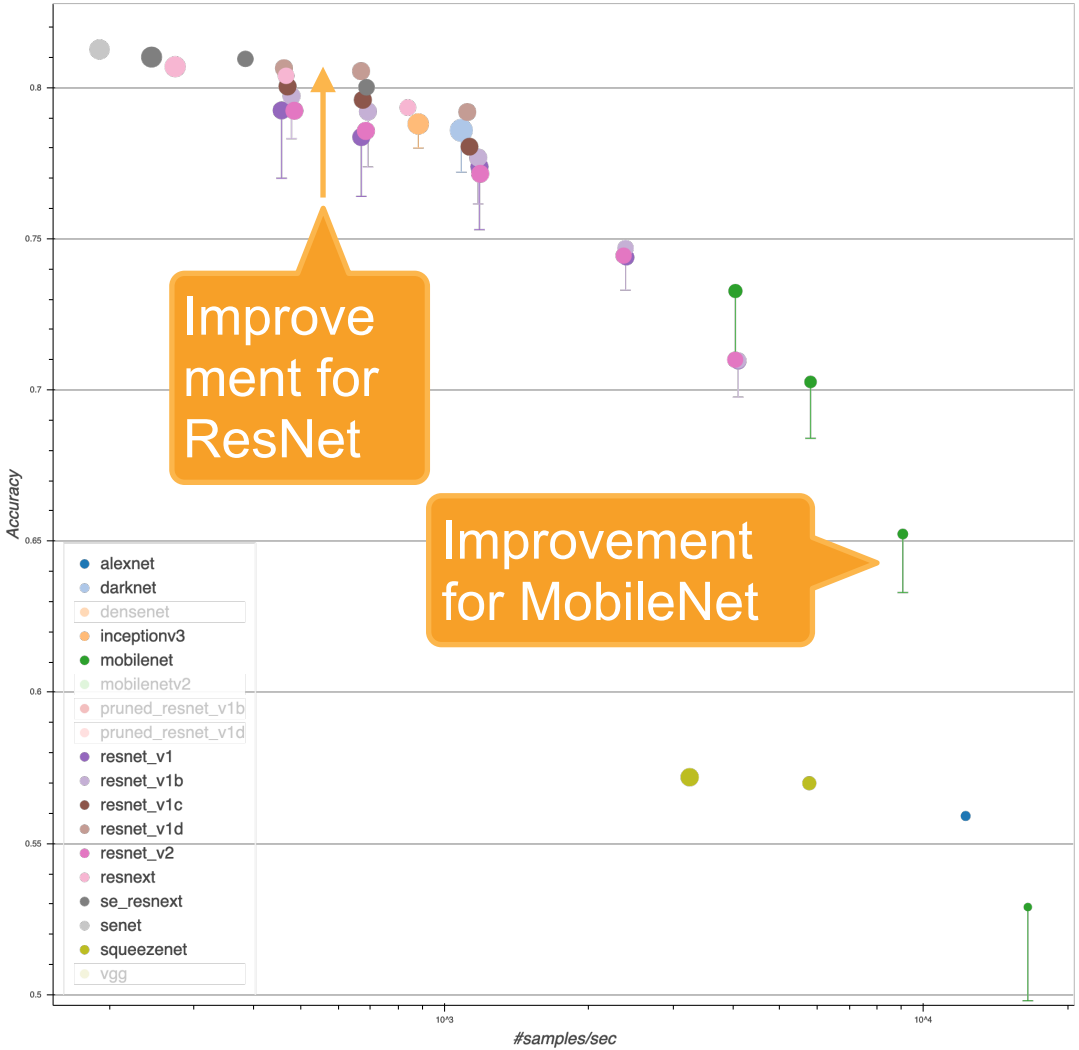
https://gluon-cv.mxnet.io/model_zoo/detection.html

http://1day-zh.d2l.ai

计算机视觉
训练技巧

http://1day-zh.d2l.ai

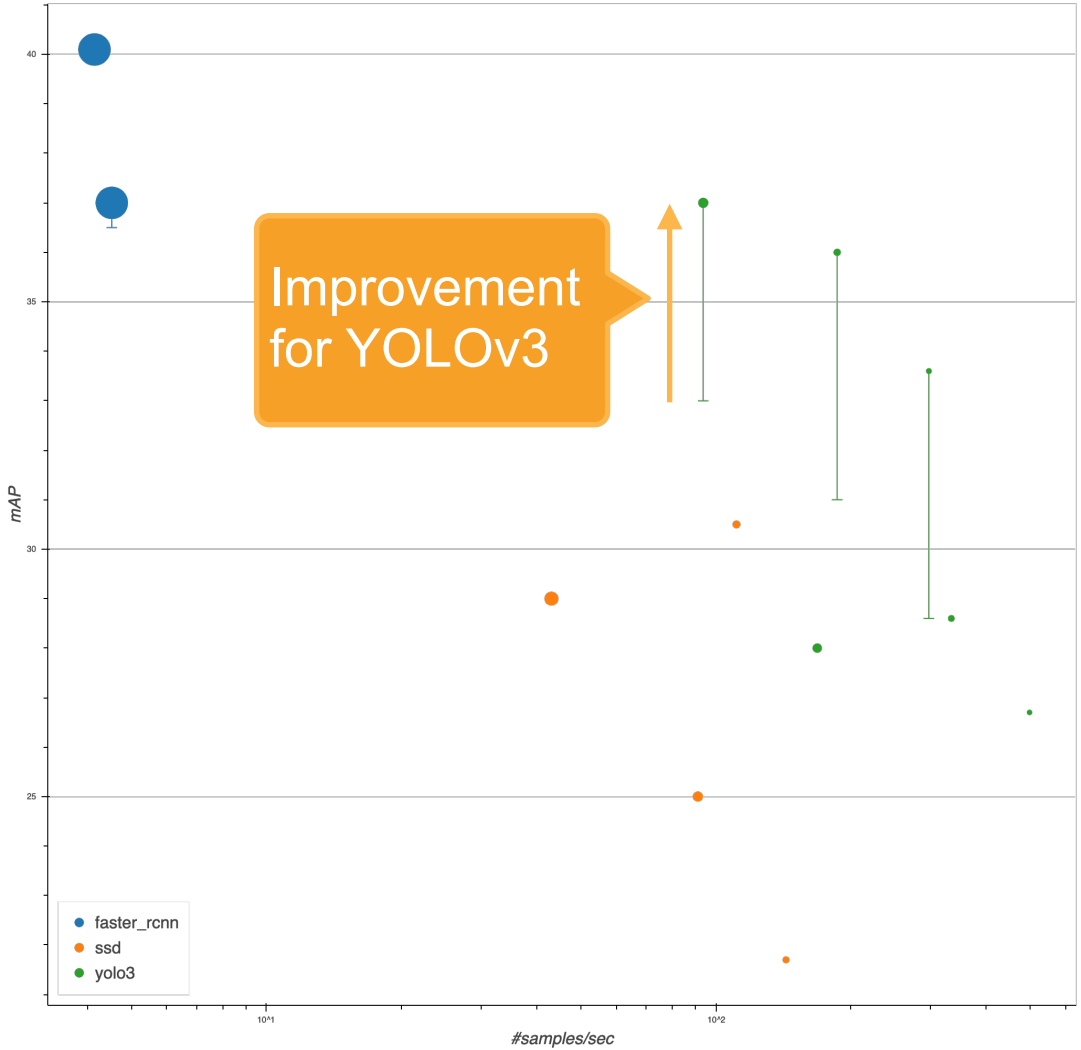
各种训练技巧可以极大提高图像分类模型的准确率



GluonCV model zoo
https://gluon-cv.mxnet.io/model_zoo/classification.html

http://1day-zh.d2l.ai

许多训练技巧也
适用于提升目标
检测模型的精度



GluonCV model zoo
https://gluon-cv.mxnet.io/model_zoo/detection.html

http://1day-zh.d2l.ai

混合训练数据

- 随机选两个图片 i 和 j， 抽取随机数 $\lambda \in [0,1]$
- 新的训练数据为 $x = \lambda x_i + (1-\lambda)x_j \quad y = \lambda y_i + (1-\lambda)y_j$
- 例如

bittern	0
...	0
otter	0
...	0
analog_clock	1

* 0.9 +

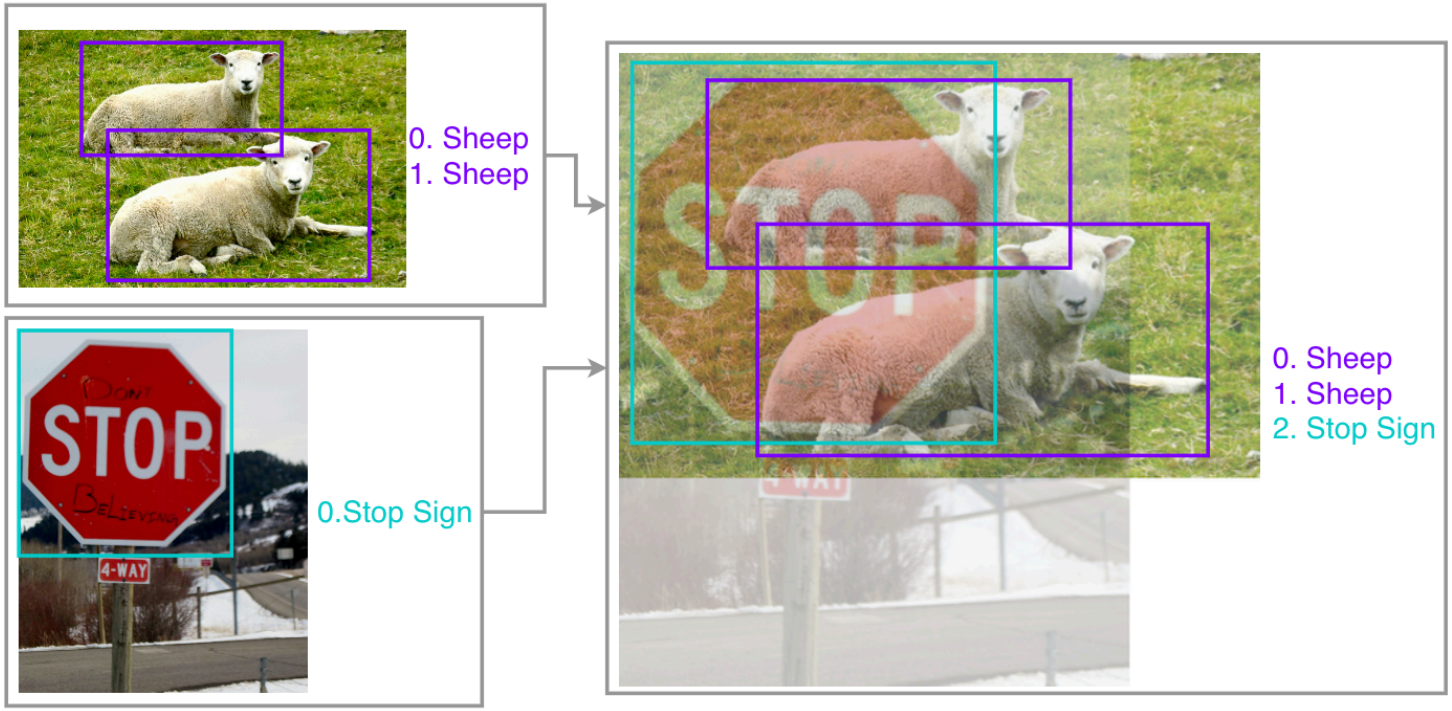
bittern	1
...	0
otter	0
...	0
analog_clock	0

* 0.1 =

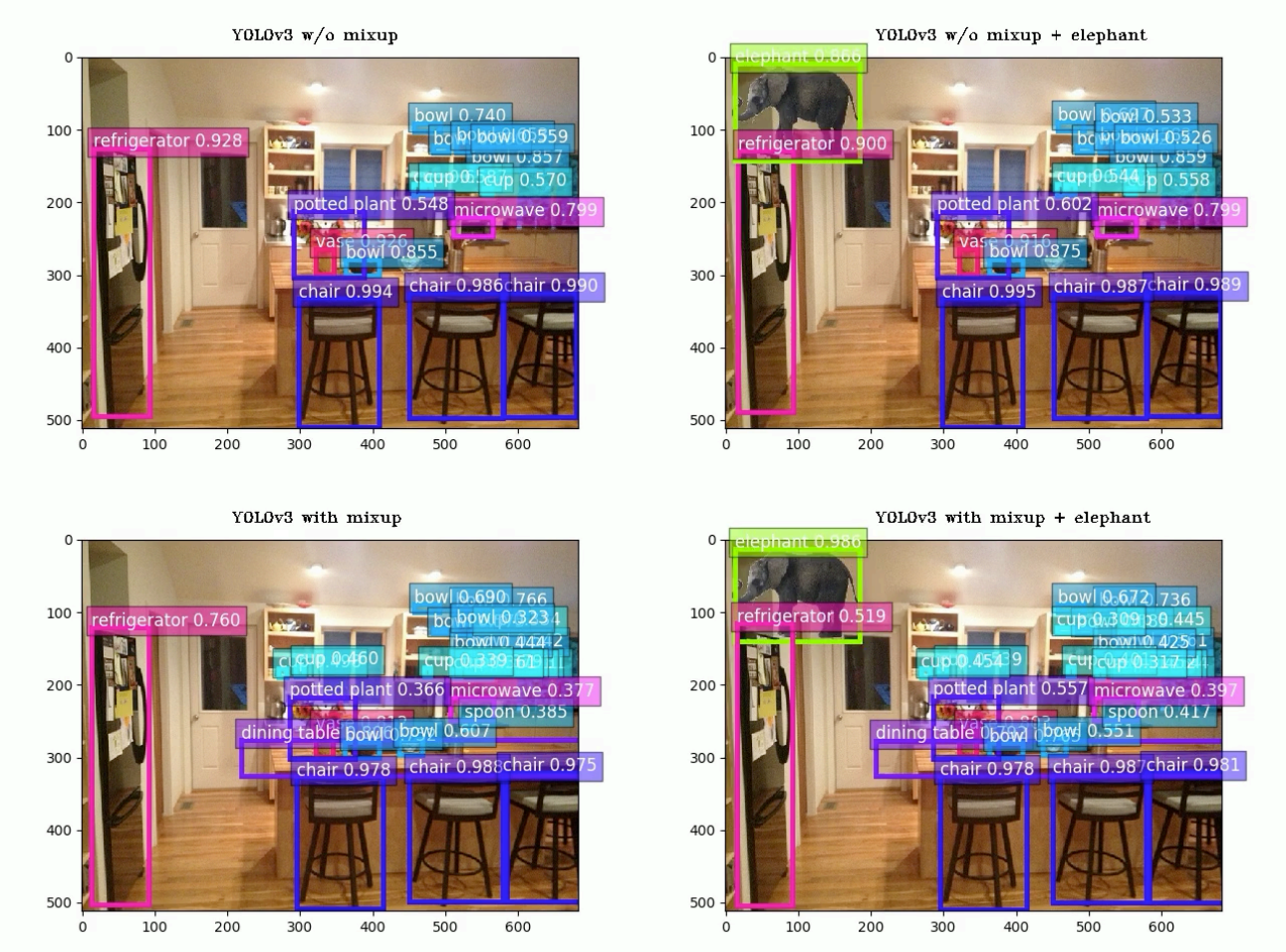
bittern	0.1
...	0
otter	0
...	0
analog_clock	0.9

混合训练数据

- 也可以用于目标检测模型



混合训练数据：更稳健



http://1day-zh.d

标签平滑化

- 假设标签 $\mathbf{y} \in \mathbb{R}^n$是单热编码(one-hot)

$$y_i = \begin{cases} 1 & \text{if belongs to class } i \\ 0 & \text{otherwise} \end{cases}$$

- 用softmax很难逼近要预测的 0/1值
- 平滑过后的标签

$$y_i = \begin{cases} 1 - \epsilon & \text{if belongs to class } i \\ \epsilon/(n-1) & \text{otherwise} \end{cases}$$

- 常设置 $\epsilon = 0.1$

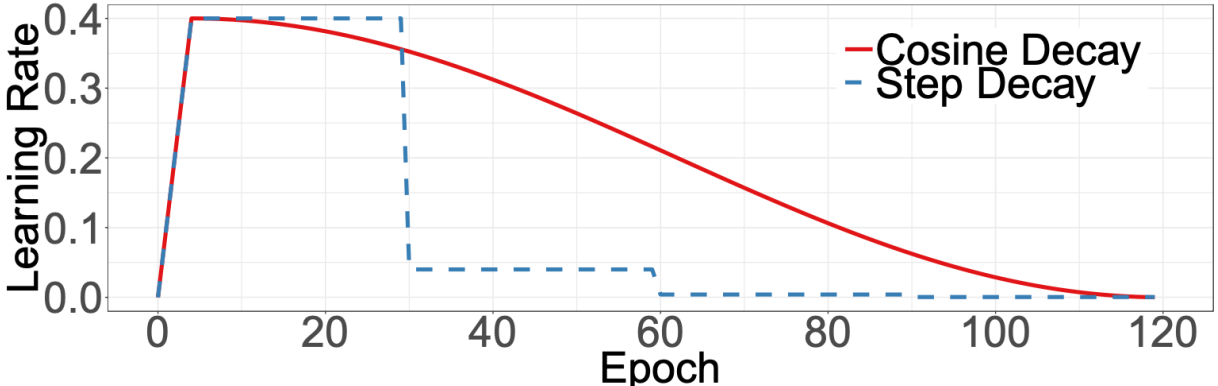
学习率预热

- 随机初始化参数应用大学习率，可能导致数值稳定性问题
- 预热技巧：在开始时使用较小的学习率，然后将其增加到预设初始学习率。
  - 如果我们选择初始学习率为0.1并使用5个周期进行预热
  - 以0开始设置学习率，在前5个周期将其线性增加到0.1

余弦衰减

- 为了收敛，我们需要降低 SGD 学习率
  - 例如在第 30, 60 和 90 个迭代周期时各减少 10 倍
- 假设在总 T 次迭代（批次）中，余弦衰减计算迭代时的学习率 t为

$$\eta_t = 1/2\left(1 + \cos\left(t\pi/T\right)\right)\eta$$



http://1day-

同步批量归一化

- BatchNorm 需要大批量才能获得可靠的统计信息
- 由于 GPU 内存限制，目标对象检测任务用小批量。
  - 例如，每 GPU只跑 1个图像
- 在多 GPU 训练中，每个 GPU 先分别计算均值/方差
- 同步批量归一化根据所有 GPU 的统计信息计算全体均值与方差

图像分类结果

Refinements	ResNet-50-D		Inception-V3		MobileNet	
	Top-1	Δ	Top-1	Δ	Top-1	Δ
Efficient	77.16		77.50		71.90	
+ cosine decay	77.91	+0.75	78.19	+0.69	72.83	+0.93
+ label smoothing	78.31	+0.4	78.40	+0.21	72.93	+0.1
+ mixup	**79 . 15**	+0.84	**78.77**	+0.37	**73.28**	+0.35

He et.al *Bag of Tricks for Image Classification with Convolutional Neural Networks*

YOLO v3 结果

Incremental Tricks	mAP	Δ	Cumu Δ
- data augmentation	64.26	-15.99	-15.99
baseline	80.25	0	0
+ synchronize BN	80.81	+0.56	+0.56
+ random training shapes	81.23	+0.42	+0.98
+ cosine lr schedule	81.69	+0.46	+1.44
+ class label smoothing	82.14	+0.45	+1.89
+ mixup	**83.68**	**+1.54**	**+3.43**

Zhi et al, *Bag of Freebies for Training Object Detection Neural Networks*

http://1day-zh.d2l.ai

总结

- 图像增广
- 微调
- 目标检测
- 边界框和锚框
- 单发多框检验（SSD）模型
- 只看一次（YOLO）
- 区域卷积神经网络（R-CNN）
- 计算机视觉训练技巧

http://1day-zh.d2l.ai